

1           **USER AUTHENTICATION METHOD, AND STORAGE MEDIUM,**  
2                   **APPARATUS AND SYSTEM THEREFOR**

3       **Field of the Invention**

4       The present invention relates to a user authentication  
5       method used, for example, for a computer system  
6       connected to a network; a storage medium on which a user  
7       authentication program is stored; a user authentication  
8       apparatus; and a user authentication system. In  
9       particular, the present invention pertains to a user  
10      authentication method, for authenticating relations  
11      existing between a prover computer, equipped with a  
12      public key, and a plurality of verifier computers; a  
13      storage medium on which such a user authentication  
14      program is stored; and a user authentication apparatus  
15      and an authentication system therefor.

16      **Background Art**

17      On a network, users are often required to participate in  
18      some sort of authentication process to identify  
19      themselves. An authentication process in this case  
20      refers to a process whereby a prover, by following the  
21      rules of a specific protocol, proves his or her identity  
22      to a verifier, a requisite electronic commerce  
23      technique. When, for example, a user desires to prove

1 his or her identity to a server, the user functions as a  
2 prover and the server functions as a verifier. Whereas  
3 when a server desires to prove its identity to a user,  
4 the server functions as a prover and the user functions  
5 as a verifier. Such authentication techniques are not  
6 limited in their application to intercourse between  
7 users and servers, but are widely employed as mutual  
8 identification methods by arbitrarily paired computers.  
9 Recently, the user authentication processes that are  
10 employed are based on public key encryption: a prover  
11 has both a public key and a secret key, and when the  
12 prover desires to prove his or her identity, he or she  
13 employs a specific protocol to notify a verifier that he  
14 or she has a secret key that corresponds to the public  
15 key.

16 The Schnorr method is a well known, representative user  
17 authentication technique ("Efficient Signature  
18 Generation by Smart Cards", C.P. Schnorr, Journal of  
19 Cryptology, Vol. 4, No. 3, pp.161-174, 1991). According  
20 to this technique, a prover proves to a verifier that he  
21 or she holds a secret key corresponding to a public key.

22 As one conventional example, a summary of Schnorr's user  
23 authentication method will now be given while referring  
24 to Fig. 3. System parameters used by this method are  
25 prime numbers  $p$  and  $q$  ( $q|p-1$ ) and the element  $g \in \mathbb{Z}_p$  of  
26 the order  $q$ . The public key of the prover is  $v$  ( $v = g^s$   
27  $\text{mod } p$ ), and the secret key of the prover is  $s \in \mathbb{Z}_q$ . In

1 the following explanation, assume that the prover and  
2 the verifier obtain in advance the prime numbers  $p$  and  $q$   
3 and the element  $g$ , which are system parameters, and that  
4 the verifier obtains in advance the public key  $v$  of the  
5 prover.

6 According to this method, the verifier and the prover  
7 exchange data in the following manner.

8 Step 1: The prover generates a random number  $a \in \mathbb{Z}_q$ ,  
9 calculates  $A = g^a \bmod p$ , and transmits it to the  
10 verifier.

11 Step 2: The verifier generates a random number  $b$  ( $b \in$   
12  $\mathbb{Z}_q$ ), and transmits it to the prover.

13 Step 3: The prover calculates  $c = a + bs \bmod q$ , and  
14 transmits it to the verifier.

15 Step 4: The verifier determines whether  $A = V^{bg^c} \bmod p$  is  
16 established. If this equation is established, the  
17 verifier ascertains that the identity of the prover is  
18 correct. If this equation is not established, the  
19 verifier ascertains that the identity of the prover is  
20 incorrect, and rejects the communication.

21 The Schnorr method is the most efficient of all the  
22 methods based on the discrete logarithm program, and  
23 only three communications are required. However, the  
24 safety of the communications is not guaranteed. That  
25 is, in the process of following the procedures defined  
26 in the protocol and communicating across the network,  
27 the secret key  $s$  of the prover may be revealed.

1 Therefore, the safety of such a data exchange between  
2 prover and verifier should be evaluated, i.e., the user  
3 authentication process (the exchange of messages, etc.).

4 For this evaluation, i.e., of the safety of the user  
5 authentication process, a zero-knowledge technique is  
6 well known ("The Knowledge Complexity of Interactive  
7 Proofs", S. Goldwasser, S. Micali, and C. Rackoff,  
8 Proceedings of 17th Symposium on Theory of Computing,  
9 pp. 291-304, 1985). In this instance, the zero  
10 knowledge property represents that no information  
11 concerning the secret key of the prover is revealed, and  
12 thus, when the zero knowledge property is achieved, the  
13 safety of the user authentication method is guaranteed.

14 The zero knowledge property can be achieved by a partial  
15 correction to the Schnorr authentication method ("How to  
16 prove yourself: practical solution to identification and  
17 signature problems", A. Fiat and A. Shamir, Proceedings  
18 of Crypto' 86, 1980). Specifically, when the Schnorr  
19 authentication method is corrected so that the verifier  
20 generates a random number  $b \in \{0, 1\}$  and so that the  
21 procedures in the protocol are sequentially performed  $O$   
22  $(\log q)$  times, the zero knowledge property is achieved.  
23 That is, when the subsequent protocol procedures are  
24 performed  $O(\log q)$  times, and if the verifier accepts  
25 the identity of the prover in all the performances of  
26 the protocol procedures, the identity of the prover is  
27 verified.

28 Protocol]

1 Step 1: The prover generates a random number  $a \in \mathbb{Z}_q$ ,  
2 calculates  $A = g^a \bmod p$  and transmits the random number  $A$   
3 to the verifier.

4 Step 2: The verifier generates a random number  $b \in \{0,$   
5  $1\}$ , and transmits the random number  $b$  to the prover.

6 Step 3: The prover calculates  $c = a + b s \bmod q$ , and  
7 transmits the result  $c$  to the verifier.

8 Step 4: The verifier determines whether  $A = v^b g^c \bmod p$   
9 has been established. When the equation has been  
10 established, the verifier concludes that the identity of  
11 the prover is correct. If the equation is not  
12 established, the verifier concludes that the identity of  
13 the prover is incorrect, and rejects the communication.

14 As described above, although the number of  
15 communications is increased to  $O(\log q)$ , the zero  
16 knowledge property is achieved. Besides the Schnorr  
17 method, many other user authentication methods have been  
18 proposed that achieve the zero knowledge property.

19 Problems to be Solved by the Invention]

20 However, to achieve the zero knowledge property for the  
21 conventional user authentication, it is proposed that  
22 one prover correspond to one verifier, and that the zero  
23 knowledge property will be achieved only when the prover  
24 and the verifier complete the performance of the  
25 protocol procedures using one-to-one correspondence (see  
26 Fig. 4). That is, when the prover must perform the  
27 protocol with multiple verifiers, there is no guarantee  
28 that the zero knowledge property will be achieved

1 ("Concurrent Zero-Knowledge", C. Dwork, M. Naor and A.  
2 Sahai, Proc. Of 30th STOC, 1998).

3 For example, on an asynchronous network, such as the  
4 Internet, multiple computers simultaneously communicate  
5 with each other, and a prover may also be required to  
6 simultaneously perform the protocol procedures with  
7 multiple verifiers. On the WWW (the World Wide Web), an  
8 HTTP (Hyper Text Transfer Protocol: the protocol used by  
9 WWW servers and WWW browsers or Web browsers to exchange  
10 such data as files) server is requested to verify its  
11 identity through simultaneous communication exchanges  
12 with multiple connected clients (see Fig. 5)

13 Summary of the Invention

14 To resolve the above shortcoming, it is one object of  
15 the present invention to provide a user authentication  
16 method whereby, even when multiple verifiers are in  
17 simultaneous communication with a prover, a user can be  
18 safely authenticated while at the same time the zero  
19 knowledge property is achieved, as well as a storage  
20 medium on which such a user authentication program is  
21 stored, and a user authentication apparatus and a user  
22 authentication system therefor.

23 To achieve the above object, according to one aspect of  
24 the present invention, a user authentication method,  
25 whereby a one-way function  $F$ , which should satisfy  $v =$

1 F(g, -s), is determined by employing an integer g that  
2 is defined in advance for a relation between a public  
3 key v and a secret key s of a prover computer, and  
4 whereby a relation is verified between the prover  
5 computer and each of multiple verifier computers,  
6 comprises the steps of: the prover computer generating a  
7 random number a, obtaining a cryptogram A = the function  
8 F(g, a), and transmitting the cryptogram A to the  
9 verifier computers; the verifier computers generating a  
10 random number b, obtaining a cryptogram B = the function  
11 F(g, b) and a cryptogram X = the function F(A, b), and  
12 transmitting the cryptograms B and X to the prover  
13 computer; the prover computer determining whether a  
14 relation of the cryptogram X = the function F(B, a) has  
15 been established and generating a random number c when  
16 the relation has been established, obtaining a  
17 cryptogram C = the function F(g, c) and a cryptogram Y =  
18 the function F(B, c), or a cryptogram C = the function  
19 F(A, c), a cryptogram Y = the function F(X, c) and a  
20 cryptogram Z = a function H(a, Y, s), and transmitting  
21 the cryptograms C and Y or the cryptograms C, Y and Z to  
22 the verifier computers; and the verifier computers, when  
23 the cryptogram Y = the function F(C, b) and the  
24 cryptogram A = a function J(v, Y, g, Z) are established,  
25 determining that the relation between the prover  
26 computer and the verifier computer is correct.

27 The public key  $v$  is obtained by employing prime numbers  
28  $p$  and  $q$  that satisfy  $(q|p - 1)$ , and by defining an  
29 element of the order  $q$  as the integer  $g$ .

1 By using the public key  $v$  and the secret key  $s$ , the  
2 function  $F$  acquires a relation  $v = F(g, -s) = g^{-s} \bmod p$ .

3 When a relation  $X = B^a \bmod p$  is established, the prover  
4 computer generates the random number  $c$ .

5 The function  $H$  has a relation  $H(a, Y, s) = a + Ys \bmod q$ .

6 The function  $J$  has a relation  $J(v, Y, g, Z) = v^Y g^Z \bmod p$ .

7 According to another aspect of the invention, a storage  
8 medium is provided on which a user authentication  
9 program, which is to be read by a prover computer, is  
10 stored whereby a one-way function  $F$ , which should  
11 satisfy  $v = F(g, -s)$ , is determined by employing an  
12 integer  $g$ , which is defined in advance for the relation  
13 between a public key  $v$  and a secret key  $s$  of the prover  
14 computer, and whereby a relation is verified between the  
15 prover computer and each of multiple verifier computers,  
16 the user authentication program permitting the prover  
17 computer to perform: a process for generating a random  
18 number  $a$  and for obtaining a cryptogram  $A =$  the function  
19  $F(g, a)$ , and for transmitting the cryptogram  $A$  to the  
20 verifier computers; a process for receiving cryptograms  
21  $B$  and  $X$  from the verifier computer, and for employing  
22 the cryptograms to determine whether a relation a  
23 cryptogram  $X =$  the function  $F(B, a)$  has been  
24 established; a process for generating a random number  $c$   
25 when the relation has been established; and a process  
26 for obtaining a cryptogram  $C =$  the function  $F(g, c)$  and  
27 a cryptogram  $Y =$  the function  $F(B, c)$ , or a cryptogram  $C$



1 = the function  $F(A, c)$ , a cryptogram  $Y$  = the function  
2  $F(X, c)$  and a cryptogram  $Z$  = the function  $H(a, Y, s)$ ;  
3 and a process for transmitting the cryptograms  $C$  and  $Y$ ,  
4 or  $C, Y$  and  $Z$ , to the verifier computers.

5 According to an additional aspect of the present  
6 invention, a storage medium is provided on which is  
7 stored a user authentication program, which is to be  
8 read by a prover computer, whereby a one-way function  $F$ ,  
9 which should satisfy  $v = F(g, -s)$ , is determined by  
10 employing an integer  $g$ , which is defined in advance for  
11 the relation between a public key  $v$  and a secret key  $s$   
12 of the prover computer, and whereby a relation is  
13 verified between the prover computer and each of  
14 multiple verifier computers, the user authentication  
15 program permitting the verifier computers to perform: a  
16 process for receiving a cryptogram  $A$  from the prover  
17 computer and for generating a random number  $b$ ; a process  
18 for obtaining a cryptogram  $B$  = the function  $F(g, b)$  and  
19 a cryptogram  $X$  = the function  $F(A, b)$ , using the random  
20 number  $b$  and the cryptogram that is received, and for  
21 transmitting the cryptograms  $B$  and  $X$  to the prover  
22 computer; a process for receiving, from the prover  
23 computer, a cryptogram  $C$  = the function  $F(g, c)$  and a  
24 cryptogram  $Y$  = the function  $F(B, c)$ , or a cryptogram  $C$  =  
25 the function  $F(A, c)$ , a cryptogram  $Y$  = the function  $F(X,$   
26  $c)$  and a cryptogram  $Z$  = the function  $H(a, Y, s)$ ; and a  
27 process, based on the cryptograms  $C$  and  $Y$  or  $C, Y$  and  $Z$   
28 that are received, for verifying a relation between the  
29 verifier computer and the prover computer when two

1 relations of the cryptogram  $Y = \text{the function } F(C, b)$  and  
2 the cryptogram  $A = \text{the function } J(v, Y, g, Z)$  are  
3 established at the same time.

4 According to a further aspect of the present invention,  
5 a user authentication apparatus is provided for a prover  
6 computer, wherein a one-way function  $F$ , which should  
7 satisfy  $v = F(g, -s)$ , is determined by employing an  
8 integer  $g$ , which is defined in advance, for a relation  
9 between a public key  $v$  and a secret key  $s$  of the prover  
10 computer, and wherein a relation is verified between the  
11 prover computer and each of multiple verifier computers,  
12 the user authentication apparatus comprising:  
13 transmission means, for generating a random number  $a$  and  
14 obtaining a cryptogram  $A = \text{the function } F(g, a)$ , and for  
15 transmitting the obtained cryptogram  $A$  to the verifier  
16 computers; reception means, for receiving cryptograms  $B$   
17 and  $X$  from the verifier computers; verification means,  
18 for employing the cryptograms  $B$  and  $X$  to determine  
19 whether a relation of the cryptogram  $X = \text{the function}$   
20  $F(B, a)$  has been established; cryptogram computation  
21 means, for generating a random number  $c$  when it has been  
22 ascertained that the relation has been established, and  
23 for obtaining a cryptogram  $C = \text{the function } F(g, c)$  and  
24 a cryptogram  $Y = \text{the function } F(B, c)$ , or a cryptogram  $C$   
25  $= \text{the function } F(A, c)$ , a cryptogram  $Y = \text{the function}$   
26  $F(X, c)$  and a cryptogram  $Z = \text{the function } H(a, Y, s)$ ;  
27 and cryptogram transmission means, for transmitting the  
28 cryptograms  $C$  and  $Y$  or  $C, Y$  and  $Z$  to the verifier  
29 computers.

1 According to a still further aspect of the prevent  
2 invention, a user authentication apparatus is provided  
3 for a prover computer wherein a one-way function  $F$ ,  
4 which should satisfy  $v = F(g, -s)$ , is determined by  
5 employing an integer  $g$ , which is defined in advance, for  
6 the relation between a public key  $v$  and a secret key  $s$   
7 of a prover computer, and wherein a relation is verified  
8 between the prover computer and each of multiple  
9 verifier computers, the user authentication apparatus  
10 comprising: reception means, for receiving a cryptogram  
11  $A$  from the prover computer; transmission means, for  
12 generating a random number  $b$ , and for employing the  
13 random number  $b$  and the cryptogram  $A$  that is received to  
14 obtain a cryptogram  $B = \text{the function } F(g, b)$  and a  
15 cryptogram  $X = \text{the function } F(A, b)$ , and for  
16 transmitting the cryptograms  $B$  and  $X$  to the prover  
17 computer; cryptogram reception means, for receiving from  
18 the prover computer a cryptogram  $C = \text{the function } F(g,$   
19  $c)$  and a cryptogram  $Y = \text{the function } F(B, c)$  or a  
20 cryptogram  $C = \text{the function } F(A, c)$ , a cryptogram  $Y =$   
21  $\text{the function } F(X, c)$ , and a cryptogram  $Z = \text{the function}$   
22  $H(a, Y, s)$ ; and verification means, for performing a  
23 procedure, based on the cryptograms  $C$ ,  $Y$  and  $Z$  that are  
24 received, for verifying a relation between the verifier  
25 computers and the prover computer when two relations of  
26 the cryptogram  $Y = \text{the function } F(C, b)$  and the  
27 cryptogram  $A = \text{the function } J(v, Y, g, Z)$  are  
28 established at the same time.

1 According to yet one more aspect of the present  
2 invention, a user authentication system comprises: the  
3 above described user authentication apparatus for the  
4 prover computer; and a plurality of the above described  
5 user authentication apparatuses for the verifier  
6 computers.

7 According to yet another aspect of the present  
8 invention, a user authentication system, wherein a  
9 one-way function  $F$ , which should satisfy  $v = F(g, -s)$ ,  
10 is determined by employing an integer  $g$ , which is  
11 defined in advance, for the relation between a public  
12 key  $v$  and a secret key  $s$  of a prover computer, and  
13 wherein a relation is verified between the prover  
14 computer and each of multiple verifier computers,  
15 comprises: transmission means, for the prover computer,  
16 for generating a random number  $a$  and obtaining a  
17 cryptogram  $A = \text{the function } F(g, a)$ , and for  
18 transmitting the obtained cryptogram  $A$  to the verifier  
19 computers; reception means for the verifier computers,  
20 for receiving the cryptogram  $A$  from the prover computer;  
21 transmission means for the verifier computers, for  
22 generating a random number  $b$  with which the cryptogram  $A$   
23 is employed to obtain a cryptogram  $B = \text{the function } F(g,$   
24  $b)$  and a cryptogram  $X = \text{the function } F(A, b)$ , and for  
25 transmitting the cryptograms  $B$  and  $X$  to the prover  
26 computer; reception means for the prover computer, for  
27 receiving the cryptograms  $B$  and  $X$  from the verifier  
28 computers; verification means for the prover computer,  
29 for employing the cryptograms  $B$  and  $X$  to determine

1 whether a relation of the cryptogram  $X$  = the function  
2  $F(B, a)$  has been established; cryptogram computation  
3 means for the prover computer, for generating a random  
4 number  $c$  when it is ascertained that the relation has  
5 been established, and for obtaining the cryptogram  $C$  =  
6 the function  $F(g, c)$  and the cryptogram  $Y$  = the function  
7  $F(B, c)$ , or the cryptogram  $C$  = the function  $F(A, c)$  and  
8 the cryptogram  $Y$  = the function  $F(X, c)$ , and a  
9 cryptogram  $Z$  = the function  $H(a, Y, s)$ ; and cryptogram  
10 transmission means for the prover computer, for  
11 transmitting the cryptograms  $C, Y$  and  $Z$  to the verifier  
12 computers; cryptogram reception means, for the verifier  
13 computers, for receiving the cryptograms  $C, Y$  and  $Z$  from  
14 the prover computer; and verification means for the  
15 verifier computers, for employing the cryptograms  $C, Y$   
16 and  $Z$  that are received to verify a relation between the  
17 verifier computers and the prover computer when two  
18 relations of the cryptogram  $Y$  = the function  $F(C, b)$  and  
19 the cryptogram  $A$  = the function  $J(v, Y, g, Z)$  are  
20 established at the same time.

#### 21 Preferred Embodiment

22 The preferred embodiment of the present invention will  
23 now be described while referring to the accompanying  
24 drawings. In this embodiment, the invention is applied  
25 for a case wherein a public key  $v$  and a secret key  $s$  are  
26 used for user authentication on a network.

27 The present invention relates to user authentication for

1 an asynchronous network, such as the Internet. In the  
2 asynchronous network, multiple verifiers may request a  
3 prover to execute a protocol for user authentication.  
4 That is, in this embodiment, there are multiple  
5 verifiers for one prover.

6 In this embodiment, the following one-way function F is  
7 employed as an encryption function. Assume that the  
8 one-way function F is a two-input and one-output  
9 function, and that two calculations, addition (+) and  
10 multiplication (\*) are defined by the range and a second  
11 variable range of a function.

12 Further, the function F satisfies the following two  
13 properties.

14 That is, for arbitrary an a and b, the following  
15 relations must be established:

16 
$$(1) F(g, a+b) = F(g, a) * F(g, b)$$

17 
$$(2) \text{ if } A = F(g, a), F(g, a*b) = F(A, b).$$

18 Another encryption function H, which is a three-input  
19 and one-output function, is represented as follows.

20 
$$H(a, Y, s) = a + Y*s$$

21 wherein the addition and multiplication are the ones  
22 defined in the second variable range of the function F.  
23 Furthermore, an additional encryption function J, which  
24 is a four-input and one-output function, is represented  
25 as follows using the function F.

26 
$$J(v, Y, g, Z) = F(v, Y) * F(g, Z).$$

27 The one-way function based on the discrete logarithm can  
28 be a specific example for the function F. As a typical

1 example, when a relation  $q|p-1$  is established for prime  
2 numbers  $p$  and  $q$  and when  $g \in \mathbb{Z}_p$  is the element of the  
3 order  $q$ ,  
4  $F(g, a) = g^a \bmod p$ .

5 A system for which the present invention can be applied  
6 is shown in Fig. 2. A prover computer 10 and a verifier  
7 computer 40, which include at the least a CPU, and  
8 additional verifier computers 60 having the same  
9 configuration as the verifier computer 40 are connected  
10 to a network 32. As is shown in Fig. 2, in this  
11 embodiment, a one-to-multiple connection is established  
12 between the prover computer and the verifier computers.

13 The prover computer 10 includes an input device 12, for  
14 entering system parameters, is connected to a random  
15 number generator 14, for generating a random number  $a$  in  
16 accordance with the input, and a memory 16. The random  
17 number generator 14 is connected to the memory 16 and a  
18 cryptogram calculator 18, for obtaining a cryptogram  $A$   
19 based on the random number  $a$ . The cryptogram calculator  
20 18 is connected to a communication interface  
21 (hereinafter referred to as a communication I/F) 30,  
22 which in turn is connected to the network 32, to  
23 facilitate communications with other apparatuses via the  
24 network 32. A verification unit 20 is connected both to  
25 the communication I/F 30 and to the memory 16. A random  
26 number generator 22, for generating a random number  $c$  in  
27 accordance with the input, and a halting unit 24, for  
28 employing an input signal to halt a protocol that will





1 sections are employed.

2 The protocol for this embodiment will now be described.  
3 It should be noted that the system parameter is a  
4 function  $F_g$ , the public key of a prover is  $v = F(g, -s)$ ,  
5 and the secret key of the prover is  $s$ .

## 6 Protocol

### 7 Step 1:

8 A prover generates the random number  $a$  using the random  
9 number generator 14, obtains a cryptogram  $A = F(g, a)$   
10 using the cryptogram calculator 18, and transmits the  
11 cryptogram  $A$  to verifiers via the communication I/F 30.  
12 Step 1 corresponds to a process  $Ps1$ , which is performed  
13 by the prover computer 10 in Fig. 1, and communication  
14  $T1$ , which is transmitted as a result of the process  $Ps1$ .

### 15 Step 2:

16 The verifier generates the random number  $b$  using the  
17 random number generator 44, and employs the received  
18 cryptogram  $A$  to obtain a cryptogram  $B = F(g, b)$  and a  
19 cryptogram  $X = F(A, b)$ . The verifier then transmits the  
20 obtained cryptograms  $B$  and  $X$  to the prover via the  
21 communication I/F 30.

22 Step 2 corresponds to a process  $Qs1$ , which is performed  
23 after the verifier computer 40 in Fig. 1 has received  
24 the data accompanying the communication  $T1$ , and to  
25 communication  $T2$ , which is transmitted as a result of  
26 the process  $Qs1$ .

1 Step 3:  
2 Based on the received cryptograms B and X, the prover  
3 employs the verification unit 20 to determine whether  $X$   
4  $= F(B, a)$  has been established for the verifier. If  $X =$   
5  $F(B, a)$  has not been established for the verifier, the  
6 prover ascertains that the verifier performed an illegal  
7 activity, and halts the performance of the protocol  
8 procedures using the halting unit 24. If, however,  $X =$   
9  $F(B, a)$  has been established for the verifier, the  
10 prover generates the random number  $c$  and obtains  $C =$   
11  $F(g, c)$  and  $Y = F(B, c)$ , or alternately, obtains  $C =$   
12  $F(A, c)$  and  $Y = F(X, c)$ . Afterwards,  $Z = H(a, Y, s)$ ,  
13 i.e.,  $Z = a + Y*s$  is calculated, and then the obtained  
14 cryptograms  $C, Y$  and  $Z$  are transmitted to the verifier.  
15 Step 3 corresponds to a process Ps2, which is performed  
16 after the prover computer 10 in Fig. 1 has received the  
17 data accompanying the communication T2, and to  
18 communication T3, which is transmitted because the  
19 relation  $X = F(B, a)$  was verified by the verification  
20 unit 20 during the process Ps2.

21 Step 4:  
22 Based on the received cryptograms  $C, Y$  and  $Z$ , the  
23 verifiers uses the verification unit 50 to determine  
24 whether  $Y = F(c, b)$  and  $A = J(v, Y, g, Z)$ , i.e.,  $A =$   
25  $F(v, Y)*F(g, Z)$ , have been established. If the two  
26 relations have been established, the verifier accepts  
27 the identity of the prover (the acceptance unit 52 is  
28 activated). If, however, the two relations have not

1 been established, the verifier rejects the identity of  
2 the prover (the rejection unit 54 is activated).  
3 Step 4 corresponds to a process Qs2 performed after the  
4 verifier computer 40 in Fig. 1 has received the data  
5 accompanying the communication T3.

6 The above protocol can be stored as a program, for use  
7 by the prover and the verifiers, on a storage medium,  
8 such as a floppy disk. In this case, only a detachable  
9 floppy disk unit (FDU) need be connected to the  
10 individual computers to enable the program to be read  
11 from the floppy disk and executed.

12 A processing program may be stored (installed) in a RAM,  
13 or at another storage area (e.g., on a hard disk) in the  
14 computer, and executed, or it may be stored in a ROM in  
15 advance. A storage medium, a disk such as a CD-ROM, an  
16 MD, an MO or a DVD, or a magnetic tape such as a DAT,  
17 may also be used, but when one of these media is  
18 employed, a corresponding device, such as a CD-ROM  
19 drive, an MD drive, an MO drive, a DVD drive or a DAT  
20 drive must be provided.

21 Specific Example:

22 A specific example of user authentication for which the  
23 above described protocol is employed will now be  
24 described. In the following example, when prime numbers  
25  $p$  and  $q$  ( $q|p - 1$ ) and the element  $g$  of the order  $q$  are  
26 employed as system parameters,  $v = F(g, -s) = g^{-s} \bmod p$   
27 is employed as the function  $F$ . That is, the same key

1 configuration as that provided by the Schnorr method can  
2 be employed. Further, the function H is defined as  $H(a,$   
3  $Y, s) = a + Y s \bmod q$ , and the function J is defined as  
4  $J(v, Y, g, Z) = v^Y g^Z \bmod p$ .

5 Key configuration]

6 System parameters: prime numbers p and q ( $q|p - 1$ ) and  
7 the element g of the order q

8 Public key of a prover:  $v = g^{-s} \bmod p$

9 Secret key of a prover:  $s \in \mathbb{Z}_q$

10 Protocol]

11 Step 1: The prover generates the random number a,  
12 acquires a cryptogram A and transmits the cryptogram A  
13 to the verifier.

14  $a \in \mathbb{Z}_q$  ... (1)

15  $A = g^a \bmod p$  ... (2)

16 That is, at the prover computer 10, the random number  
17 generator 14 employs the system parameter q to generate  
18 the random number a, in accordance with expression (1),  
19 and the cryptogram calculator 18 employs the random  
20 number a and the system parameters p and q to obtain the  
21 cryptogram A, in accordance with expression (2). The  
22 obtained cryptogram A is then output through the  
23 communication I/F 30, and is transmitted, via the  
24 network 32, to the verifier computer 40.

25 Step 2: The verifier generates the random number b,  
26 obtains cryptograms B and X, and transmits the  
27 cryptograms B and X to the prover.



- 1  $C = g^c \text{ mod } p \quad \dots(8)$
- 2  $Y = B^c \text{ mod } p \quad \dots(9)$
- 3 or  $C = A^c \text{ mod } p \quad \dots(10)$
- 4  $Y = X^c \text{ mod } p \quad \dots(11)$
- 5  $Z = a + Y s \text{ mod } q \quad \dots(12)$

6 Specifically, at the prover computer 10 the verification  
7 unit 20 receives the cryptograms B and X from the  
8 verifier computer 40 via the communication I/F 30, and  
9 employs the cryptograms B and X that are received and  
10 the system parameters stored in the memory 16 to examine  
11 the cryptograms B and X, in accordance with expression  
12 (6).  
13 If expression (6) has not been established, the  
14 verification unit 20 transmits a signal to the halting  
15 unit 24 to halt the performance of the protocol  
16 procedures. When expression (6) has been established,  
17 however, the verification unit 20 outputs a signal to  
18 the random number generator 22 to generate the random  
19 number c at the random number generator 44 based on the  
20 system parameter q, following which the random number c  
21 is transmitted to the cryptogram calculator 26, which  
22 employs the random number c, the received cryptogram B  
23 and the system parameters p and g to obtain cryptograms  
24 C and Y, in accordance with expressions (8) and (9), or  
25 (10) and (11). Then, in accordance with expression  
26 (12), the cryptogram calculator 26 obtains a cryptogram  
27 Z using the obtained cryptogram Y, the random number a,  
28 the secret key s and the system parameter q, and  
29 thereafter, the cryptograms C, Y and Z are output  
30 through the communication I/F 30, and are transmitted,

1 via the network 32, to the verifier computer 40.

2 Step 4: The verifier determines whether the following  
3 expressions (13) and (14) have been established. If the  
4 two expressions have been established, the verifier  
5 accepts the identity of the prover. Otherwise, the  
6 verifier rejects the identity of the prover.

7  $Y = C^b \bmod p \quad \dots(13)$

8  $A = v^Y g^Z \bmod p \quad \dots(14)$

9 Specifically, in the verifier computer 40, the  
10 verification unit 50 receives the cryptograms C, Y and Z  
11 from the prover computer 10 via the communication I/F  
12 56. Then, in accordance with expressions (13) and (14),  
13 the verification unit 50 examines the cryptograms C, Y  
14 and Z using the system parameters stored in the memory  
15 46.

16 When expressions (13) and (14) have not been  
17 established, the verification unit 50 activates the  
18 rejection unit 54 to reject the identity of the prover.  
19 When, however, the expressions (13) and (14) have been  
20 established, the verification unit 50 activates the  
21 acceptance unit 52 to accept the identity of the prover.

22 In this embodiment, user authentication can be completed  
23 through the exchange of only three communications by the  
24 prover and the verifier, and the quantity of the  
25 communications contributes to the prime numbers p and q.  
26 According to this embodiment, the number of  
27 communications is  $|p|$ , using the cryptogram A  
28 accompanying communication T1,  $2|p|$ , using the

1 cryptograms B and X accompanying communication T2, and  
2  $2|p|$  and  $|q|$ , using the cryptograms C, Y and Z  
3 accompanying communication T3 (see Fig. 1). Therefore,  
4 a total of only  $5|p| + |q|$  communications is required.  
5 Further, as is apparent from the above expressions, this  
6 contributes greatly to the reduction of the load imposed  
7 by the calculation of powers. Since only six such  
8 calculations are required, an efficient protocol is  
9 provided.

10 In this example, communication between one prover and a  
11 single verifier (one verifier) has been employed.  
12 However, on an asynchronous network, such as the  
13 Internet, the authentication of the identity of a prover  
14 must be accomplished by multiple verifiers. In this  
15 embodiment, when individual verifiers are in any of the  
16 communication states corresponding to communication T1  
17 to communication T3 (see Fig. 1), secrecy can be  
18 maintained; a secret key will not be compromised even  
19 when the cryptograms A, B, C, X, Y and Z that are  
20 transmitted are trapped en route and analyzed. This  
21 will be explained later in detail. Therefore, even when  
22 multiple verifiers must simultaneously or sequentially  
23 be permitted to examine the identity of a prover, the  
24 user authentication process can be precisely performed  
25 for each of the multiple verifiers. Thus, when multiple  
26 verifiers are permitted to examine the identity of a  
27 prover via an asynchronous network, such as the  
28 Internet, the user authentication process can be  
29 performed safely.



1 In the above example, the power calculation for  $Z_p$  is  
2 employed as a specific one-way function  $F$ , and is a  
3 so-called one-way function based on a discrete  
4 logarithm. However, the present invention is not  
5 limited to this problem; while  $N$  is a composite number,  
6 the discrete logarithm for  $Z_N$  may be employed, or the  
7 discrete logarithm for an elliptic curve may be  
8 employed.

9 Validity of protocol]

10 The validity of the protocol for this embodiment will  
11 now be described. Specifically, an explanation will be  
12 given based on the above Specific example wherein it is  
13 shown that the zero knowledge property is achieved, even  
14 when the protocol for this embodiment is applied for an  
15 asynchronous network. Whereas it is well known that the  
16 zero knowledge property is not achieved when the  
17 protocol mentioned in the description of the background  
18 art ("Concurrent Zero-Knowledge", C. Dwork, M. Naor and  
19 A. Shai, Proc. Of 30th STOC, 1998) is applied for an  
20 asynchronous network.

21 On an asynchronous network, a plurality of illegal  
22 verifiers ( $V_1, V_2, \dots$  and  $V_n$ ) may enter into a  
23 conspiracy with each other to communicate with a prover  
24  $P$ . Therefore, it is not sufficient to consider the  
25 achievement of the zero knowledge property for  
26 communications between a prover  $P$  and a single verifier  
27  $V$ . In other words, the zero knowledge property for  
28 communications between a prover  $P$  and multiple verifiers

1 V1 to Vn must be taken into account.

2 In the authentication process in this embodiment, it is  
3 proved that the information that can be obtained through  
4 communication, in accordance with the proposed protocol,  
5 with the prover P by multiple illegal verifiers V1 to  
6 Vn, who have entered into a conspiracy with each other,  
7 can be obtained without the communication with the  
8 prover P. Specifically, it is proved for arbitrary  
9 illegal verifiers V1 to Vn, there is an algorithm S  
10 (simulator) such that the probability distribution of  
11 the output of S matches the one of the contents of the  
12 actual communications exchanged by the prover P and each  
13 verifier V1 to Vn. In this embodiment, this proof is  
14 represented as "the algorithm S simulates the contents  
15 of the actual communication between the prover P and  
16 each verifier V1 to Vn".

17 Conspiracy of verifiers]

18 It may be assumed that, without losing generality, the  
19 illegal verifiers V1 to Vn in a conspiracy communicate  
20 with the prover P in the following manner. The  
21 verifiers V1 to Vn are sorted into groups G1, G2, ...  
22 and Gm ( $m \leq n$ ). Intuitively, it is assumed that a  
23 verifier who belongs to the group G<sub>i</sub> communicates with  
24 the prover P based on information obtained by a verifier  
25 who belongs to the group G<sub>i-1</sub>.

26 Generalized conspiracy protocol]

27 The input data are employed as the public key for the

1 prover P and as the system parameters  $(p, q, g, v)$ .

2 Step 1: The prover P calculates cryptograms  $A_1 = g^{a_1}$ ,  $A_2$   
3  $= g^{a_2}$ , ... and  $A_n = g^{a_n} \bmod p$ , and transmits the obtained  
4 cryptograms  $A_1, A_2, \dots$  and  $A_n$  to the respective  
5 verifiers  $V_1, V_2, \dots$  and  $V_n$ .

6 The information obtained by the verifiers  $V_1$  to  $V_n$  is  
7  $VIEW_0 = \{(p, q, g, v), (A_1, A_2, \dots, A_n)\}$ .

8 Step 2-1-P: All the verifiers  $V_i$  who belong to the group  
9  $G_1$  employ the received cryptograms  $A_1$  to  $A_n$  to generate  
10 a random number  $b_i \in \mathbb{Z}_q$ , and obtain cryptograms  $B_i (= g^{b_i}$   
11  $\bmod p)$  and  $X_i (= A_i^{b_i} \bmod p)$ . The verifiers  $V_i$  then  
12 transmit the obtained cryptograms  $B_i$  and  $X_i$  to the  
13 prover P.

14 Step 2-1-V: The prover P examines each  $i$  that satisfies  
15  $V_i \in G_1$  to determine whether the authentication  
16 expression  $(X_i = B_i^{a_i} \bmod p)$  has been established.  
17 If the authentication expression has been established,  
18 the prover P transmits the cryptograms  $C_i, Y_i$  and  $Z_i$  to  
19 the verifiers  $V_i$ .  
20 At this time, the information obtained by the verifiers  
21 is  $VIEW_1 = VIEW_0 \cup \{(B_i, X_i, C_i, Y_i, Z_i) \mid V_i \in G_1\}$ .

22 Then, steps 2-k-P and 2-k-V are repeated for  $2 \leq k \leq n$ .

23 Step 2-k-P: All the verifiers  $V_i$  who belong to the group  
24  $G_k$  employ the obtained information  $VIEW_{k-1}$  to generate a

1 random number  $b_i \in \mathbb{Z}_q$ , and obtain cryptograms  $B_i (= g^{b_i}$   
2  $\text{mod } p)$  and  $X_i (= A_i^{b_i} \text{mod } p)$ . The verifiers  $V_i$  then  
3 transmit the obtained cryptograms  $B_i$  and  $X_i$  to the  
4 prover  $P$ .

5 Step 2-k-V: The prover  $P$  examines each  $i$  that satisfies  
6  $V_i \in G_k$  to determine whether the authentication  
7 expression  $(X_i = B_i^{a_i} \text{mod } p)$  has been established.  
8 If the authentication expression has been established,  
9 the prover  $P$  transmits the cryptograms  $C_i$ ,  $Y_i$  and  $Z_i$  to  
10 the verifiers  $V_i$ .

11 At this time, the information obtained by the verifiers  
12 is  $\text{VIEW}_k = \text{VIEW}_{k-1} \cup \{(B_i, X_i, C_i, Y_i, Z_i) \mid V_i \in G_k\}$ .

13 As a result, the information finally obtained by the  
14 verifiers who are members of the conspiracy is

15  $\text{VIEW}_n = \{(p, q, g, v),$   
16  $(A_1, A_2, \dots, A_n),$   
17  $(B_1, B_2, \dots, B_n),$   
18  $(X_1, X_2, \dots, X_n),$   
19  $(C_1, C_2, \dots, C_n),$   
20  $(Y_1, Y_2, \dots, Y_n),$   
21  $(Z_1, Z_2, \dots, Z_n)\}.$

22 Assumption of calculation amount for conspiracy]  
23 In order to establish  $x_i = B_i^{a_i} \text{mod } p$  for each  $i$  at the  
24 step 2-k-V, the verifiers  $V_i$  use a random number  $b_i \in \mathbb{Z}_q$   
25 to calculate  $B_i = g^{b_i} \text{mod } p$  and  $X_i = A_i^{b_i} \text{mod } p$ . In other  
26 words, it is presumed that each verifier  $V_i$  knows the

1 value of the random number  $b_i$ . This assumption can be  
2 formally described as follows.

3 b-awareness assumption: hereinafter referred to as BAA]

4 At steps 2-1-V, 2-2-V, ... and 2-n-V, relative to an  
5 arbitrary verifier  $V_i$ , there is another verifier  $V_i'$  who  
6 outputs not only the cryptograms  $B_i$  and  $X_i$ , but also  
7 outputs the value of the random number  $b_i$ .

8 Configuration of simulator]

9 When the simulator  $S$  is constructed as follows, the zero  
10 knowledge property can be achieved under the BAA. The  
11 simulator  $S$  employs the verifiers ( $V_1'$ ,  $V_2'$ , ... and  
12  $V_n'$ ) as sub-routines, and can thus employ the individual  
13 random numbers  $b_i$ .

14 Algorithm of simulator]

15 Input: public key  $v$ , system parameters  $p$ ,  $q$  and  $g$

16 Output:  $VIEW_n = \{(p, g, g, v),$   
17  $(A_1, A_2, \dots, A_n),$   
18  $(B_1, B_2, \dots, B_n),$   
19  $(X_1, X_2, \dots, X_n),$   
20  $(C_1, C_2, \dots, C_n),$   
21  $(Y_1, Y_2, \dots, Y_n),$   
22  $(Z_1, Z_2, \dots, Z_n)\}$

23 Step 1: For all "i"s ( $1 \leq i \leq n$ ), random numbers  $Y_i \in Z_q$   
24 and  $Z_i \in Z_q$  are generated, and  $A_i = V^{Y_i}g^{Z_i}$  is calculated.

1 At this time, the simulation information produced by the  
2 simulator S is  
3  $VIEW_0 = [(p, q, g, v), (A_1, A_2, \dots, A_n)]$ .  
4 Step 2-1-P: The simulator S executes all the verifiers  
5  $V_i$  ( $V_i'$ ) who belong to the group  $G_1$ . That is,  $VIEW_0$  is  
6 input for each verifier  $V_i'$ , and  $(B_i, X_i, b_i)$  are  
7 calculated. At this time,  $B_i = g^{b_i} \text{ mod } p$  is established.  
8 Step 2-1-V:  $C_i$  that satisfies  $Y_i = C_i^{b_i} \text{ mod } p$  is  
9 calculated. At this time, the simulation information  
10 produced by the simulator S is  
11  $VIEW_1 = VIEW_0 \cup \{(B_i, X_i, C_i, Y_i, Z_i) \mid V_i \in G_1\}$ .  
12 Then, steps 2-k-P and 2-k-V are repeated for  $2 \leq k \leq n$ .  
13 Step 2-k-P: The simulator S executes all the verifiers  
14  $V_i$  ( $V_i'$ ) who belong to the group  $G_k$ . That is,  $VIEW_{k-1}$  is  
15 input to each verifier  $V_i'$ , and  $(B_i, X_i, b_i)$  are  
16 calculated. At this time,  $B_i = g^{b_i} \text{ mod } p$ .  
17 Step 2-k-V:  $C_i$  that satisfies  $Y_i = C_i^{b_i} \text{ mod } p$  is  
18 calculated. At this time, the information simulated by  
19 the simulator S is  $VIEW_k = VIEW_{k-1} \cup \{(B_i, X_i, C_i, Y_i,$   
20  $Z_i) \mid V_i \in G_k\}$ .  
21 The communication contents  $VIEW_n$ , which are finally to be  
22 simulated, match the probability distribution of the  
23 actual communication contents between the prover P and  
24 the verifiers  $V_1, V_2, \dots$  and  $V_n$ . Therefore, the zero  
25 knowledge property is achieved.

00010350-032501

1 Advantages of the Invention]

2 As is described above, according to the present  
3 invention, the secret key of a prover computer is not  
4 compromised by the information exchanged by the prover  
5 computer and a verifier computer, and user  
6 authentication is ensured.  
7 Especially when on an asynchronous network, such as the  
8 Internet, a prover computer receives data required for  
9 authentication as well as verification from multiple  
10 verifiers, the zero knowledge property is acquired.  
11 Thus, user authentication is ensured without the secret  
12 key of a prover computer being compromised on any kind  
13 of network.

14 The present invention can be realized in hardware,  
15 software, or a combination of hardware and software. The  
16 present invention can be realized in a centralized fashion  
17 in one computer system, or in a distributed fashion where  
18 different elements are spread across several  
19 interconnected computer systems. Any kind of computer  
20 system - or other apparatus adapted for carrying out the  
21 methods described herein - is suitable. A typical  
22 combination of hardware and software could be a general  
23 purpose computer system with a computer program that, when  
24 being loaded and executed, controls the computer system  
25 such that it carries out the methods described herein. The  
26 present invention can also be embedded in a computer  
27 program product, which comprises all the features enabling  
28 the implementation of the methods described herein, and

1 which - when loaded in a computer system - is able to  
2 carry out these methods.

3 Computer program means or computer program in the present  
4 context mean any expression, in any language, code or  
5 notation, of a set of instructions intended to cause a  
6 system having an information processing capability to  
7 perform a particular function either directly or after  
8 conversion to another language, code or notation and/or  
9 reproduction in a different material form.

10 It is noted that the foregoing has outlined some of the  
11 more pertinent objects and embodiments of the present  
12 invention. This invention may be used for many  
13 applications. Thus, although the description is made for  
14 particular arrangements and methods, the intent and  
15 concept of the invention is suitable and applicable to  
16 other arrangements and applications. It will be clear to  
17 those skilled in the art that other modifications to the  
18 disclosed embodiments can be effected without departing  
19 from the spirit and scope of the invention. The  
20 described embodiments ought to be construed to be merely  
21 illustrative of some of the more prominent features and  
22 applications of the invention. Other beneficial results  
23 can be realized by applying the disclosed invention in a  
24 different manner or modifying the invention in ways known  
25 to those familiar with the art.